FloReMi: Flow density survival Regression using Minimal feature redundancy

Sofie Van Gassen[1,2,3], Celine Vens[2,3,4], Tom Dhaene[1], Bart N. Lambrecht[2,3], Yvan Saeys[2,3]

[1] Department of Information Technology, Ghent University - iMinds, Ghent, Belgium

[2] Inflammation Research Center, VIB, Ghent, Belgium

[3] Department of Respiratory Medicine, Ghent University Hospital, Ghent, Belgium

[4] Department of Public Health and Primary Care, KU Leuven Kulak, Kortrijk, Belgium

Running headline:

FloReMi

Contact information:

Sofie Van Gassen
E-mail: sofie.vangassen@irc.vib-ugent.be
Address: VIB-Ghent University, Technologiepark 927, B-9052 Ghent (Zwijnaarde), Belgium
Telephone number: +32 9 331 38 10
Fax number: +32 9 221 76 73

Credits:

Sofie Van Gassen is funded by a Ph.D. grant of the Agency for Innovation by Science and Technology in Flanders (IWT). Celine Vens is a Postdoctoral Fellow of the Research Foundation - Flanders (FWO-Vlaanderen). This work was supported by the Ghent University Multidisciplinary Research Partnership Bioinformatics: from nucleotides to networks.

Abstract

Advances in flow cytometry bioinformatics have resulted in a wide variety of clustering, classification and visualization techniques. To objectively evaluate the performance of such methods, common benchmarks such as the FlowCAP initiative have proven to be of great value. In this work, we report on a novel method, FloReMi, which was developed to tackle the most recent FlowCAP IV challenge.

This challenge was formulated as a survival modeling problem, where participants were expected to design a model to predict the time until progression to AIDS for HIV patients. It is known that variability in progression rate cannot be fully predicted by simple $CD4^+$ T cell counts. However, it is hypothesized that the immunopathogenesis established early in HIV already indicates the course of future disease. Adequately estimating the progression rate of HIV patients is crucial in their treatment.

Using an automated pipeline to preprocess the data, and subsequently identify and select informative cell subsets, a survival regression method based on random survival forests was built, which obtained the best results of all submitted approaches to the FlowCAP IV challenge.

Our pipeline is available at www.github.com/SofieVG/FloReMi.

Key terms

polychromatic flow cytometry; machine learning; survival time prediction; feature selection

Introduction

Current cytometry techniques enable researchers to examine many markers at the same time. This gives an unprecedented view on single cells, but also introduces several challenges. When many markers are measured simultaneously, manual analysis becomes very time-consuming and subjective. As it is infeasible to manually analyze every possible cell population, only a subset of them is examined based on previous experience. Several research groups have developed automatic clustering techniques to assist the manual analysis, based on K-means, mixture models, density estimation and more (e.g. 1-8). However, the goal of an experiment is often not to identify all the cell types that are present, but to identify those cell types that are indicative of some phenotype. In this case, machine learning techniques can be used to identify subpopulations of cells in the dataset which can be used to predict the phenotype of a patient.

A number of techniques have already been developed with this goal in mind. The FlowCAP II challenge (9) was created to compare those techniques and provide some benchmark data. Several algorithms were proposed, most of which combine an automatic clustering algorithm with a traditional classification algorithm. Once the clusters are formed, features can be constructed and selected, which can be used by traditional classification algorithms. One of the proposed algorithms, FlowType (10,11), computes a threshold for every marker to express every possible subpopulation as a certain combination of high/low marker values. Once all these subtypes are defined, statistical tests are performed to identify those subtypes which contain information with regard to the class of the patients. The Citrus algorithm (12) uses a hierarchical clustering algorithm to split the dataset into many possible subdivisions. Afterwards, statistical tests are used to find significant differences between the data of two classes. Classification algorithms trained with those features can be used to diagnose patients.

The majority of algorithms that were proposed in this field focus on classification tasks, where a class, e.g. a disease status, is assigned to each patient. Less attention has been given to predict continuous output variables based on flow cytometry data. To address this issue, the FlowCAP IV challenge was proposed. The goal of the FlowCAP IV challenge was to predict the time until progression to AIDS for HIV patients, a task which was manually studied in (13). In this paper, we present our approach for the FlowCAP IV challenge, combining the flowType algorithm with a feature selection algorithm to identify informative, non-redundant features. We evaluated three survival time prediction algorithms using the selected features, of which the random survival forest approach was the most successful, and obtained the best predictive performance of all methods submitted to the challenge.

Problem definition

The goal of the FlowCAP IV challenge was to predict the time until progression to AIDS for a test set of 192 HIV patients, based on a training set of 191 patients. Patients were described by flow cytometry data, from which features needed to be extracted to reach this goal. Identifying features that correlate with the time until progression to AIDS, e.g. the size of a specific cell population in the dataset, was also a goal of this challenge.

For each patient, a PBMC sample stimulated with HIV-Gag peptides and an unstimulated control were provided. The unstimulated sample gives an indication of the baseline state of the patient, whereas in the stimulated sample immune response effects to the antigens might be observed. For each sample, FSC-A, FSC-H, SSC-A and 13 fluorescence channels were measured (indicating values for IFNγ, TNFα, CD4, CD27, CD107-A, CD154, CD3, CCR7, IL2, CD8, CD57, CD45RO and V-Amine/CD14). Each patient of the training set was also assigned a label indicating the observed clinical status (1 = progression to AIDS or death, 0 = no progression to AIDS or death) and, if there was progression to AIDS or death, the survival time until the onset of AIDS. If there was no progression to AIDS, the time to the last evaluation was given. It is important to notice that a label of zero does not mean that the patient did not develop AIDS, but only that the patient at least did not develop AIDS until this last observation. This kind of data is called 'censored' information, because what happens after the last evaluation is unknown. From the 191 patients in the training set, only 34 had actual events and 157 were censored. In the test set, 45 patients had actual events and 147 were censored. This strongly complicated the computational framework we had to use to build a model for this data.

The evaluation criterion for the challenge was based on the Cox proportional-hazards model (14). This model uses both event data and censored data to study the dependency of the survival times on predictor variables and results in a p-value indicating how well the variables fit the survival times. During evaluation, the predicted survival times are used as predictor variable.

The FloReMi algorithm

The FloReMi approach consists of 4 steps. First the data is preprocessed, in order to remove noise. Subsequently many features (i.e. properties pertaining to certain cell types) are extracted, after which a selection of these features is made. Finally, we use the selected features in a regression model to predict the time until the detection of AIDS for the patients. A schematic overview of our approach is given in Figure 1. Our method was developed specifically for the FlowCAP IV challenge, but our scripts are available at www.github.com/SofieVG/FloReMi and can be adapted for other datasets.

Preprocessing

The preprocessing step was applied to each sample separately and consisted of 6 parts.

We started with a quality control step to detect problems during the acquisition of the sample by the cytometer. This can be done by inspecting uniformity of the data with respect to the time parameter (15). Therefore, we split the dataset in 100 equally sized intervals for the Time parameter. We calculated the median FSC-A value and the number of cells for each interval. Intervals were removed completely if

either their median FSC-A value differed more than 10,000 from the interval right before or after it, or if the number of cells in the interval was less than the median number of cells per interval minus two standard deviations. These thresholds were defined after inspecting several problematic sample files. By removing these intervals, we removed measurements with inconsistent values, caused by e.g. disturbances of the flow stream, air bubbles or clogging of the flow cell. A similar technique has been proposed by Fletez-Brant, which is released in the flowClean R bioconductor library (16). On average, 5.30% (±3.60%) of the original cells were removed by this step

In the next preprocessing part, we removed all margin events. We defined a margin event as a measurement which had either the minimum or maximum value in any dimension, or a measurement which exceeded the ranges given in the description of the fcs-files. Measurements which exceeded the possible ranges were erroneous, and measurements which had the minimum or maximum value might be saturated and out of the detection range of the cytometer, thus not representing the actual value that should be measured. A similar technique is available in the flow cytometry module of GenePattern: RemoveSaturatedFCSEvents (17). On average, 2.72%(±3.83%) of the cells remaining after the first preprocessing step were removed by this step.

Our third preprocessing step consisted of removing doublets. We computed the ratio *r* between FSC-A and FSC-H, since for doublets, the area measured is larger in proportion to the height, because the signal has the same strength but a longer duration in comparison with a single cell (18). We removed all cells for which the ratio was larger than the median ratio of all cells plus two standard deviations, on average 4.45%(±1.25%) of the remaining cells after the first two preprocessing steps.

$$r_{cell\_to\_keep} \leq median(r_{all\_cells}) + 2\ stdev(r_{all\_cells})$$

The fourth and fifth steps in our preprocessing procedure were the traditional flow cytometry preprocessing steps: compensation and transformation. We compensated the data using the spillover matrix provided in the fcs-files. We transformed the SSC channel and all color channels with the logicleTransform() function from the R flowCore package (19), using all default settings.

Our final preprocessing step helped us to zoom in on the area of interest. We used an automatic gating step to select only the alive T-cells for further analysis. To do this, we used the R flowDensity package (20). This package can automatically determine an optimal split in a single dimension of the dataset. We used this to determine thresholds for both the V450-A (Vivid/CD14) and R780-A (CD3) channels. We selected those cells that were low for V450-A (alive, no macrophages or monocytes) and high for R780-A (T cells).

Feature extraction

Once the dataset was clean, we extracted features from it. By using automatic, unsupervised techniques, we were able to examine a much larger scope of features than would be possible in any manual analysis. The feature extraction part of the pipeline was executed on each sample separately.

First, we used the flowDensity algorithm (20) again to determine splits on ten dimensions: FSC-A, SSC-A, G710-A (CD4), G660-A (CD27), G610-A (CD107-A), G560-A, (CD154), R710-A (CCR7), V800-A (CD8), V585-

A (CD57), V545-A (CD45RO). This algorithm uses the density distribution of the cells to determine the best possible split. If two peaks are detected, the minimum intersection point between the two peaks is used. If there are more peaks, it takes into account the distance between the peaks and the height of the valleys to determine which split gives the clearest cut. If no peaks are detected, it will use the 95^{th} percentile to split on. We excluded the FSC-H, V450-A and R780-A channels because they were dealt with in the preprocessing step. We also excluded the intracellular markers IFNγ, IL2 and TNFα, in order to reduce computation time, even though this might lead to a loss of information. These intracellular markers do not have two clear peaks typically, which makes it harder to get a good split automatically.

The second step in the feature extraction process was to define subsets, groups of cells which have the same annotation of high/low marker intensities, based on the thresholds determined by the flowDensity algorithm. We did this by using the flowType algorithm (10), examining every possible marker combination for which the values are either high, low or neutral. By exploring all cell subsets with combinations of high and low expression of the markers, we included many possible cell types that might not be identified in a manual analysis. The flowType bioconductor package provides an efficient implementation to assign cells to the subpopulations they belong to, using dynamic programming as an optimization for the combinatorial problem (11). This led to 3^{10} possibilities or 59,049 different subsets.

Once each subset was defined, we extracted features for each sample. For each subset, we computed the percentage of cells and the mean fluorescence intensity for 13 markers (FSC-A, SSC-A, B515-A (IFNγ), G780-A (TNFα), G710-A (CD4), G660-A (CD27), G610-A (CD107-A), G560-A (CD154), R710-A (CCR7), R660-A (IL2), V800-A (CD8), V585-A (CD57), V545-A (CD45RO)). Because we included the intracellular markers IFNγ, TNFα and IL2, their information might still be used even though they were not included in the subset definitions. This leads to 14 features per subset. Because there was both stimulated and unstimulated data present for each patient, we computed those fourteen features for all subsets for both samples, and also the difference between the corresponding features from each sample. This resulted in (3^{10} * 14) *3 or 2,480,058 features per patient.

Feature selection

Regression techniques are not able to efficiently handle such a huge amount of features. To solve this problem, the third step of our pipeline was a feature selection step. In this step, we wanted to select those features which have a high correlation with the survival time. However, because of the high percentage of censored values, we could not simply use the Pearson correlation to measure the importance of a feature. Therefore, we made use of the Cox proportional-hazards model.

We used the whole training dataset to compute a Cox proportional-hazards model for each feature separately. This model returned a p-value and a concordance index, which indicated how strongly the feature itself corresponded with the actual survival times of the patients. We ranked the features by the p-values from their corresponding models.

Strongly correlated features will have a negative impact on a Cox proportional-hazards model. In order to minimize the redundancy between the selected features, we therefore did not simply pick the first *k* features from the resulting list. Instead, we started with the two features with the lowest p-values, and

then iteratively added new features when the pairwise Pearson correlation between the previously selected features and the new candidate feature was low enough. We chose a threshold of 0.2, to make sure that no strong correlations exist between the selected features. When using different thresholds, such as 0.15 or 0.25, different features were selected (Supplementary Tables 1 and 2). However, final prediction results were very similar (Supplementary Table 3).

Survival time prediction

The selected features were used to predict the actual survival time of the patients: we converted the original training and test datasets consisting of all fcs files to two matrices in which the rows represented the patients and the columns the selected features. We evaluated three different regression techniques to reach this goal: Cox proportional-hazards regression (14), random survival forests (21) and additive hazards regression (22). For each technique, we performed leave-one-out cross validation on the training dataset to evaluate our results, and built a final model using the whole training dataset to make predictions for the test set.

Next to the p-value of the Cox proportional-hazards model, we also used the concordance index (23) to evaluate our results. To compute the concordance index, all pairs of comparable patients are checked. A pair is comparable when either both patients have progressed to AIDS, or one patient has progressed to AIDS in a shorter time than the time to the last evaluation of the censored patient. The concordance index is the percentage of pairs for which the predicted survival time order corresponds to the actual order. This means a random assignment will lead to a score of about 0.5, whereas a perfect assignment will give a score of 1 and a reverse assignment will give a score of 0.

The first technique we executed was the Cox proportional-hazards regression. We built this model with an increasing number of uncorrelated features until the corresponding concordance index did no longer improve, as illustrated in Figure 2. This resulted in a feature set of 13 features, as presented in Table 3. It might be surprising that no TNFα related features are included in this set. However, notice that we start with the best scored features and only add those with almost no correlation (<0.2) with the features already selected. Some other features with better p-values and a correlation greater than 0.2 with the TNFα related features are added first to the selection, which results in no TNFα related features in our end selection. E.g. our third selected feature, the difference of the CD107a MFI values of the SSC- CD27+ CD107a+ CD154- CD8+ CD45RO- cell subset, has a correlation of 0.25 with the best ranking TNFα related feature (the TNFα MFI of the stimulated sample of the SSC+ CD4- CD27+ CD107a- CCR7- CD57- CD45RO- cell subset).

The second technique we evaluated was the random survival forest, as implemented by the randomForestSRC package(24). The random survival forest uses survival trees, in which each split is made in such a way that it maximizes the survival difference between the daughter nodes. It explicitly takes censored data into account. We used the same 13 features as we had used with the Cox proportional-hazards regression and trained a forest with 500 regression trees. This model returns the mortality, rather than the survival time, of patients. To report our results, we scaled the values between 0 and 1 and reversed them, because a higher mortality corresponds with a shorter time until progression to AIDS or death.

Finally, we also used regularization for semiparametric additive hazards regression, as implemented by the ahaz package (25). This method is a regularized version of the standard hazards model, and thus should inherently perform feature selection (similar to standard Lasso or Elastic net for the traditional regression setting). With the best 100 features from the feature selection step, a model was trained by performing a 5-fold internal cross validation on the training set to find the optimal parameter settings. These parameter settings were then used to train a final model. We also tested the model with the best 80 or 200 features, but the results were very similar (Supplementary Table 4).

Results and Discussion

During the challenge, the correct results for the test dataset were obviously hidden for the participants. Therefore, we performed leave-one-out cross validation to evaluate our techniques. We present the results of our cross-validation in Table 1. Since all three algorithms performed well for cross-validation, we submitted all three versions to the FlowCAP IV challenge.

Once the challenge was finished, the correct results for the test dataset were communicated. We present our results in Table 2. Surprisingly, the results of the three algorithms differed greatly on the test dataset. Both the Cox proportional-hazards regression and the additive hazards regression algorithm performed very badly. They have concordance scores around 0.5, which corresponds with a random assignment. However, the random survival forest algorithm did perform quite well on the test set, and actually obtained the best result of all participants of the FlowCAP IV challenge. Seven other groups participated, using a whole range of methods: one based on Boruta and FlowFP (26), one based on SPADE (27), one using a regression tree on a target vector combining clinical diagnosis and survival time, gEM/GANN (28), another also based on flowDensity and flowType, but combined with RchyOptimyx (11) and two other methods of which the strategy is not known to us at the moment.

Our results are illustrated in Figure 3, where the real survival times are compared to the predicted scores. For the training dataset, a clear correlation between the predictions and the real survival times is present for all algorithms: the line has an upward slope and if it takes more time for an event to occur, the patient will get a higher score. Notice that the steepness of the curve is not informative because we rescaled all our results between 0 and 1 at the end of the prediction process. However, on the test set, we see the same results as described above: there is no correlation at all for the results of the Cox proportional-hazards model and the additive hazards model. For the random survival forest, there is some correlation, but the data spread is much wider: many patients get a score corresponding to their survival time, but several patients get a score that is too high. We want to stress the importance of our feature selection step here. If we pick the 1000 features with the best scores, without our redundancy-based selection process, the concordance index of the Random Survival Forest model drops to 0.512 for the test set.

Because the Cox proportional-hazards regression and the additive hazards regression performed well on the training data, but badly on the test data, we suspect overfitting might be the problem. The models do not generalize to new data, which indicates they are capturing too much specific details of the training set. When a model is used for diagnosis, the main goal is to make predictions for new samples, which are not available at the time of building the model. However, when overfitting on the training dataset, the model uses very specific details of the training set which are not applicable to the whole population of interest. As such, it will fail to make good predictions for new samples, which is the case here in the validation on the test set. In the prediction step we performed leave-one-out cross-validation, but in the feature selection step, the whole training dataset was used to score the features. The random forest approach uses the same features, but might be more robust against overfitting because of the ensemble approach. Intuitively, random forests may also perform better because they are a non-linear model and, as such, they are better able to model interactions between features.

All our results were generated on a single computing node. The feature extraction step took about 3 days, while all other parts took only a couple of minutes, so this part is clearly the bottleneck for a faster workflow. However, it could be easily parallelized, because the preprocessing and feature extraction happens independently for each sample. When a computing cluster is available, this could strongly reduce running time.

The goal of the challenge was not only to predict the survival times, but also to identify features that might be useful for the diagnosis of HIV patients. In Table 3, we present the 13 features that were used for the Cox proportional-hazards regression model and the random survival forest. We notice that features from the unstimulated sample, from the stimulated sample and from the differences between the two are selected, indicating that it was essential to use both samples in the analysis. The chosen features contain not only the percentage of cells for specific subpopulations, but also the mean fluorescence intensities for several markers. These features imply that a shift in abundance of markers might happen for certain cell types.

It is important to keep in mind that all population boundaries are automatically determined and might not exactly correspond with a manual gating of the data. Even if this would be the case, it is still quite hard to interpret the features in a biological way. For example, the best feature (see top row in Table 3) is negative for all five specified markers and neutral for the others, while traditionally cell types are defined in a positive way, by having a certain marker present. In general, every marker except CD4 is used as least as much as a negative marker than as a positive marker. This might indicate that a certain cell type which does not express the stained markers could correspond well with the progression rate. It is likely that our best feature corresponds with effector CD8+ cells (29), but future research will be necessary to interpret all the features correctly and gain more insight in the process from HIV to AIDS.

Conclusion

In this paper we presented the FloReMi approach for the FlowCAP IV challenge, in which we analyzed flow cytometry data to predict survival times of HIV patients. We first thoroughly cleaned the data and extracted more than 2.4 million features for each patient. A feature selection step selected relevant features with minimal redundancy.

We evaluated three different survival time prediction methods, of which only the random survival forest method performed well. This method obtained the best results in the FlowCAP IV challenge by using a selection of 13 features.

It is interesting to notice that the four steps of the FloReMi pipeline work independently of each other. A new preprocessing step, feature extraction method or feature selection method could be plugged in without any problems, and several prediction algorithms can be used, as we did in this paper. This leaves much room for improvement for each of the steps separately and poses finding an optimal combination as a new goal.

In future work, we will investigate the optimization of each step of the pipeline. Other score metrics could be used to rank the features, although they do have to take censored data into account. The Random Survival Forest computes a mortality score instead of a survival time, which are closely related but not exactly the same. Other regression techniques which can handle censored data might work better.

Once the algorithm is optimized, more research could be done to interpret the resulting features. Biological validation might be necessary and the results we present can be seen as a starting point for other research projects to unravel the details of HIV.

Acknowledgement

We would like to thank the organizers of the FlowCAP challenge for offering a platform to compare the state-of-the-art results and providing data that stimulates the development of new algorithms.

References

1. Lo K, Brinkman R and Gottardo R. Automated gating of flow cytometry data via robust model-based clustering. Cytometry A 2008; 73:321-332.

2. Sugar I, Sealfon S. Misty mountain clustering: application to fast unsupervised flow cytometry gating. BMC Bioinformatics 2010; 11(1):502.

3. Zare H, Shooshtari P, Gupta A, Brinkman R. Data reduction for spectral clustering to analyze high throughput flow cytometry data. BMC Bioinformatics 2010; 11:403.

4. Aghaeepour N, Nikolic R, Hoos HH and Brinkman RR. Rapid cell population identification in flow cytometry data. Cytometry Part A 2011; 79(1):6–13.

5. Cron A, Gouttefangeas C, Frelinger J, Lin L, Singh SK, Britten CM, Welters MJP, van der Burg SH, West M and Chan C. Hierarchical modeling for rare event detection and cell subset alignment across flow cytometry samples. PLoS computational biology 2013; 9(7), e1003130.

6. Ge Y, Sealfon S. flowPeak: A fast unsupervised clustering for flow cytometry data via K-means and density peak finding. Bioinformatics 2014; 28:2052-2058.

7. Naim I, Datta S, Rebhahn J, Cavenaugh JS, Mosmann TR and Sharma G. Swift: scalable clustering for automated identification of rare cell populations in large, high-dimensional flow cytometry datasets, part 1: Algorithm design. Cytometry Part A 2014; 85.5:408-421.

8. Pyne S, Wang K, Irish J, Tamayo P, Nazaire MD, Duong T, Ng S, Hafler D, Levy R, Nolan GP, Mesirov J and McLachlan GJ. Joint Modeling and Registration of Cell Populations in Cohorts of High-Dimensional Flow Cytometric Data. Plos One 2014; 9:e100334.

9. Aghaeepour N, Finak G, FlowCAP Consortium, DREAM Consortium, Hoos HH, Mosmann TR, Brinkman R, Gottardo R, Scheuermann RH. Critical assessment of automated flow cytometry data analysis techniques. Nature methods 2013; 10(3):228–238.

10. Aghaeepour N, Chattopadhyay PK, Ganesan A, O'Neill K, Zare H, Jalali A, Hoos HH, Roederer M and Brinkman RR. Early immunologic correlates of HIV protection can be identified from computational analysis of complex multivariate T-cell flow cytometry assays. Bioinformatics 2012; 28(7): 1009-1016.

11. O'Neill K, Jalali A, Aghaeepour N, Hoos H and Brinkman RR. Enhanced flowType/RchyOptimyx: a Bioconductor pipeline for discovery in high-dimensional cytometry data. Bioinformatics 2014; 30(9): 1329-1330.

12. Bruggner RV, Bodenmiller B, Dill DL, Tibshirani RJ and Nolan GP. Automated identification of stratifying signatures in cellular subpopulations. Proceedings of the National Academy of Sciences 2014; 111(26): E2770-E2777.

13. Ganesan A, Chattopadhyay PK, Brodie TM, Qin J, Gu W, Mascola JR, Michael NL, Follmann DA and Roederer M. Immunologic and virologic events in early HIV infection predict subsequent rate of progression. Journal of Infectious Diseases 2010; 201(2): 272-284.

14. Fox J and Weisberg S. Cox Proportional-Hazards Regression for Survival Data in R (2011). An R Companion to Applied Regression, Second Edition. Thousand Oaks CA: Sage.

15. Watson JV. Time, a quality-control parameter in flow cytometry. Cytometry 1987; 8(6):646-649.

16. Fletez-Brant K. flowClean: flowClean. R package version 1.2.0.

17. Reich M, Liefeld T, Gould J, Lerner J, Tamayo P, Mesirov JP. GenePattern 2.0. Nature Genetics 2006; 38(5):500-501.

18. Mahnke YD and Roederer M. OMIP-001: Quality and phenotype of Ag-responsive human T-cells. Cytometry Part A 2010; 77(9):819-820.

19. Ellis B, Haaland P, Hahne F, Le Meur N, Gopalakrishnan N and Spidlen J. flowCore: Basic structures for flow cytometry data. R package version 1.32.0.

20. Malek M, Taghiyar MJ, Chong L, Finak G, Gottardo R and Brinkman RR. flowDensity: reproducing manual gating of flow cytometry data by automated density-based cell population identification. Bioinformatics 2015; 31(4): 606-607.

21. Ishwaran H, Kogalur UB, Blackstone EH and Lauer MS. Random survival forests. The Annals of Applied Statistics 2008; 2(3):841-860.

22. Lin DY and Ying Z. Additive hazards regression models for survival data. Proceedings of the First Seattle Symposium in Biostatistics 1997; 185-198.

23. Harrell FE, Lee KL and Mark DB. Tutorial in biostatistics multivariable prognostic models: issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. Statistics in medicine 1996; 15: 361-387.

24. Ishwaran H and Kogalur UB. Random Forests for Survival, Regression and Classification (RF-SRC). R package version 1.5.5. (2014).

25. Gorst-Rasmussen A and Scheike TH. Coordinate Descent Methods for the Penalized Semiparametric Additive Hazards Model. Journal of Statistical Software 2012; 47(9): 1-17.

26. Rogers WT and Holyst HA. FlowFP: a bioconductor package for fingerprinting flow cytometric data. Advances in bioinformatics 2009; 10.1155/2009/193947.

27. Qiu P, Simonds EF, Bendall SC, Gibbs Jr KD, Bruggner RV, Linderman MD, Sachs K, Nolan GP and Plevritis SK. Extracting a cellular hierarchy from high-dimensional cytometry data with SPADE. Nature biotechnology 2011; 29(10), 886-891.

28. Tong DL, Ball GR and Pockley AG. gEM/GANN: A multivariate computational strategy for auto-characterizing relationships between cellular and clinical phenotypes and predicting disease progression time using high-dimensional flow cytometry data. Cytometry Part A 2015, 10.1002/cyto.a.22622.

29. Chattopadhyay PK and Roederer M. Good cell, bad cell: Flow cytometry reveals T-cell subsets important in HIV disease. Cytometry Part A 2010; 77(7): 614-622.

Tables

Cross-validation result	Cox Proportional-Hazards	Random Survival Forests	Additive Hazards
p-value	0.000	0.000	0.000
Concordance index	0.891	0.852	0.815

Table 1: Results of the three regression models on leave-one-out cross validation. All methods performed well during cross validation.

Final results	Cox Proportional-Hazards			Random Survival Forests			Additive Hazards		
	All data	Train	Test	All data	Train	Test	All data	Train	Test
p-value	0.000	0.000	**0.733**	0.000	0.000	**0.002**	0.000	0.000	**0.782**
Concordance index	0.662	0.932	**0.459**	0.813	0.976	**0.672**	0.635	0.875	**0.527**

Table 2: Final results of the three regression models on the FlowCAP IV challenge. Only the Random Survival Forest obtains good results for the test set, which might indicate overfitting for the other models.

Feature	Sample	Subset
Percentage of cells	Unstimulated	CD4- CD27- CD107a- CD154- CD45RO-
Percentage of cells	Unstimulated	CD4- CD27- CD154- CD8+
CD107a MFI	Difference	SSC- CD27+ CD107a+ CD154- CD8+ CD45RO-
Percentage of cells	Difference	FSC- CD4+ CD107a- CD154- CCR7+ CD8+ CD57- CD45RO-
CD4 MFI	Unstimulated	FSC- CCR7- CD57- CD45RO-
IL2 MFI	Difference	FSC+ SSC+ CD107a- CCR7- CD8-
IL2 MFI	Unstimulated	FSC- SSC- CD4+ CD27- CD107a+ CD8- CD57-
IFNγ MFI	Difference	CD27- CD107a- CD154+ CCR7- CD57+ CD45RO+
CD57 MFI	Unstimulated	SSC- CD4+ CD27- CD107a- CCR7- CD8+
Percentage of cells	Stimulated	FSC- CD4- CD27- CD154- CCR7+ CD8- CD57-
Percentage of cells	Stimulated	FSC+ SSC+ CD4- CD154- CCR7+ CD8- CD57- CD45RO+
CD8 MFI	Difference	CD4+ CD27- CD107a+ CCR7- CD8- CD57-
CD8 MFI	Difference	SSC- CD154+ CD57+ CD45RO-

Table 3: Overview of the top 13 features of the feature selection step, which were used by the Cox proportional-hazards model and the random survival forest. Both features from the stimulated and the unstimulated samples are used, as well as comparisons between the two. Both percentages of cells and mean fluorescence intensities (MFI) of specific markers are used. It is remarkable that mainly a negative selection of markers is used in the selected subsets.

Figures

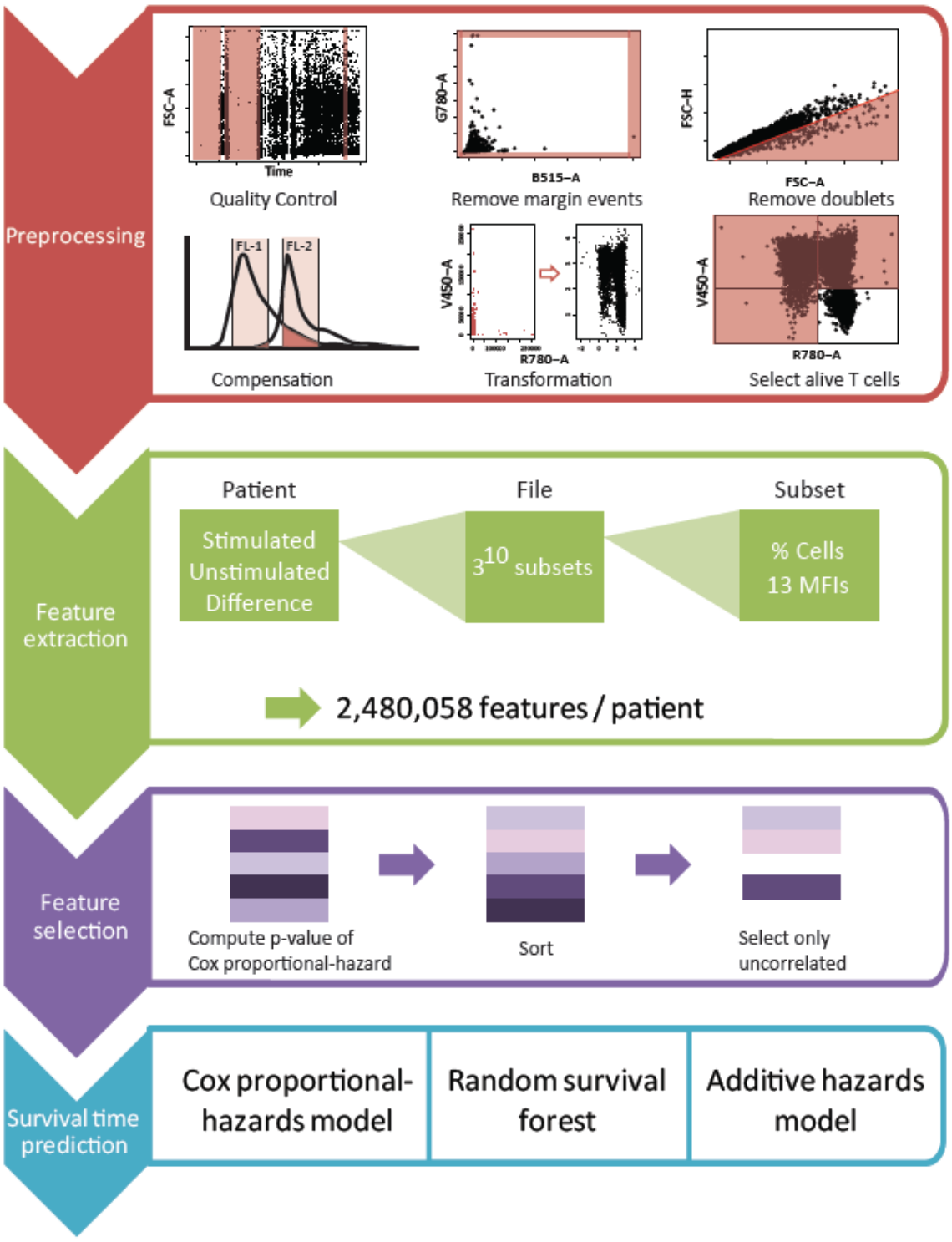
Preprocessing
FSC-A
Time
Quality Control
G780-A
B515-A
Remove margin events
FSC-H
FSC-A
Remove doublets
FL-1
FL-2
Compensation
V450-A
R780-A
Transformation
V450-A
R780-A
Select alive T cells
Feature extraction
Patient
Stimulated
Unstimulated
Difference
File
3^{10} subsets
Subset
% Cells
13 MFIs
2,480,058 features / patient
Feature selection
Compute p-value of
Cox proportional-hazard
Sort
Select only
uncorrelated
Survival time prediction
Cox proportional-hazards model
Random survival forest
Additive hazards model

Figure 1: Overview of the four steps of the FloReMi algorithm: preprocessing, feature extraction, feature selection and survival time prediction. (i) During the preprocessing, 6 steps are executed: problems during measurement are detected, margin events and doublets are removed, the data is compensated and logicle-transformed and alive T cells are selected. (ii) During feature extraction, 3^{10} subsets are identified. Each subset can be described by the percentage of cells present in the subset and 13 MFI values. All these features are computed for the stimulated and the unstimulated data, and then also the differences between the features for the two data sets are added. This results in 2,480,058 features per patient. (iii) During feature selection, a Cox proportional-hazards model is built for each feature separately, and the features are sorted by p-value (lowest first). For the actual selection, we start with the two first features and only add those which have pairwise correlation lower than 0.2 to all other selected features. (iv) In the final step, we evaluated three different survival time prediction models: the Cox proportional-hazards model, the random survival forest and the additive hazards model.

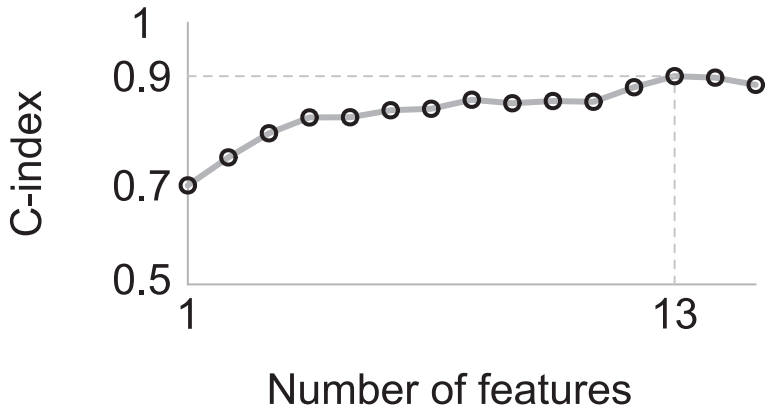

Figure 2: Concordance indices used to choose our feature selection cut-off. Using 13 features, the concordance index was optimal for the training set.

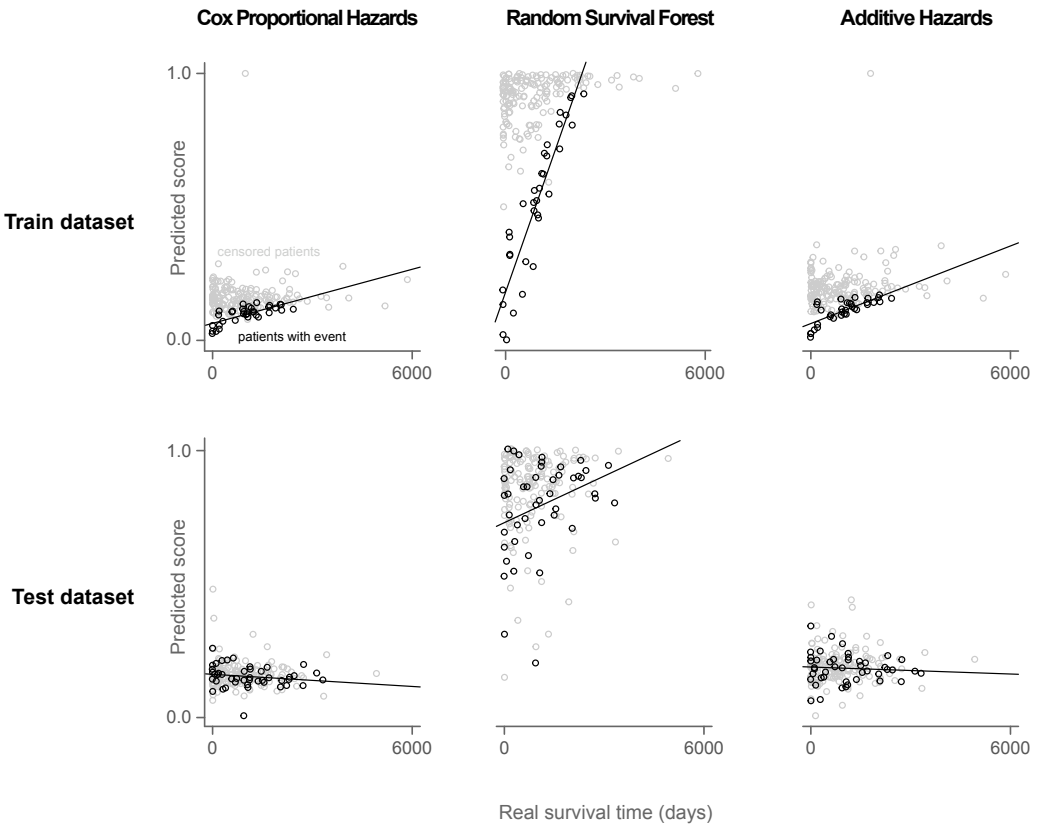

Figure 3: Results of the different models on training and test set. The prediction scores of the models have all been rescaled between 0 and 1 to provide a ranking. On the training set, all models have a correlation with the real survival times, as indicated by the regression line through the patients with events. However, on the test set, only the random survival forest has a correlation. This indicates that the other models overfit on the training set.

Supplementary Tables

Feature	Sample	Subset
Percentage of cells	Unstimulated	CD4- CD27- CD107a- CD154- CD45RO-
Percentage of cells	Unstimulated	CD4- CD27- CD154- CD8+
Percentage of cells	Difference	FSC- CD4+ CD107a- CD154- CCR7+ CD8+ CD57- CD45RO-
IFNγ MFI	Stimulated	FSC- CD4+ CD107a+ CD45RO-
IL2 MFI	Stimulated	SSC- CD107+ CCR7- CD8- CD45RO-
IL2 MFI	Difference	FSC+ SSC+ CD107a- CCR7- CD8-
Percentage of cells	Stimulated	SSC+ CD4- CD27- CD107a+ CD154- CCR7+ CD57-
CD8 MFI	Difference	CD4+ CD27- CD107a+ CCR7- CD8- CD57-
CD8 MFI	Difference	SSC- CD154+ CD57+ CD45RO-

Supplementary Table 1: Overview of the top features selected with a correlation threshold of 0.15

Feature	Sample	Subset
Percentage of cells	Unstimulated	CD4- CD27- CD107a- CD154- CD45RO-
Percentage of cells	Unstimulated	CD4- CD27- CD154- CD8+
CD4a MFI	Difference	CD4+ CD27+ CCR7- CD8- CD45RO+
CD107a MFI	Difference	SSC- CD27+ CD107a+ CD154- CD8+ CD45RO-
CCR7 MFI	Unstimulated	FSC- CD4- CD57- CD45RO-
CD4 MFI	Difference	FSC+ CD4- CD27- CD57+
IL2 MFI	Difference	SSC- CD27+ CCR7- CD57+ CD45RO+
IL2 MFI	Difference	FSC+ SSC+ CD107a- CCR7- CD8-
Percentage of cells	Stimulated	FSC- SSC- CD4- CD107a- CCR7- CD57-
Percentage of cells	Difference	FSC- SSC+ CD4- CD27- CD154- CD57-
IFNγ MFI	Difference	CD27- CD107a- CD154+ CCR7- CD57+ CD45RO+
CD8 MFI	Difference	SSC- CD4+ CD107a+ CD8- CD57-
Percentage of cells	Stimulated	FSC- CD4- CD27- CD107a+ CD154- CCR7+ CD8- CD57-

Supplementary Table 2: Overview of the top features selected with a correlation threshold of 0.25

Threshold 0.15	Cox Proportional-Hazards			Random Survival Forest		
	All data	Train	Test	All data	Train	Test
p-value	0.000	0.000	**0.737**	0.000	0.000	**0.042**
Concordance index	0.632	0.906	**0.429**	0.763	0.973	**0.572**

Threshold 0.25	Cox Proportional-Hazards			Random Survival Forest		
	All data	Train	Test	All data	Train	Test
p-value	0.000	0.000	**0.634**	0.000	0.000	**0.015**
Concordance index	0.676	0.899	**0.517**	0.828	0.973	**0.707**

Supplementary Table 3: The final prediction results are very similar for slight variations in the correlation feature selection threshold, where different features are used instead of the 13 reported ones.

Final results	Additive Hazards 80 features			Additive Hazards 100 features			Additive Hazards 200 features		
	All data	Train	Test	All data	Train	Test	All data	Train	Test
p-value	0.000	0.000	**0.832**	0.000	0.000	**0.782**	0.000	0.000	**0.578**
Concordance index	0.629	0.871	**0.540**	0.635	0.875	**0.527**	0.645	0.878	**0.478**

Supplementary Table 4: The additive hazards method is quite robust to changes in the number of input features, because it includes regularization in the model building.